PRIN Multicriteria DS 3rd meeting

Giovanni Manzini
DiSIT, Università Piemonte Orientale

Participants (as of 28/2/2021)

- Lavinia Egidi
- Giovanni Manzini
- Manuel Striani

Research activities:

- Compressed linear algebra (with unipi)
- Wheeler automata (multicriteria compressed indices?)
- Compression of learned indices (with unipi)
- PPM data compression vs NN compression (with unipi and unipa)
- Algorithms for large collections of genomes (Spire 20, Alenex 21, DCC 21)

Compressed Linear Algebra

A paper by Elgohary et al. “Compressed linear algebra for large-scale machine learning” has shown the advantages of matrix compression to speedup some machine learning computations. This paper was best paper at VLDB and featured in CACM

The key idea: data compression improves speed by reducing access costs. This is something our community has used for years. Elgohary et al. mainly used heuristics, we plan to achieve better results in a more principled way

Our approach

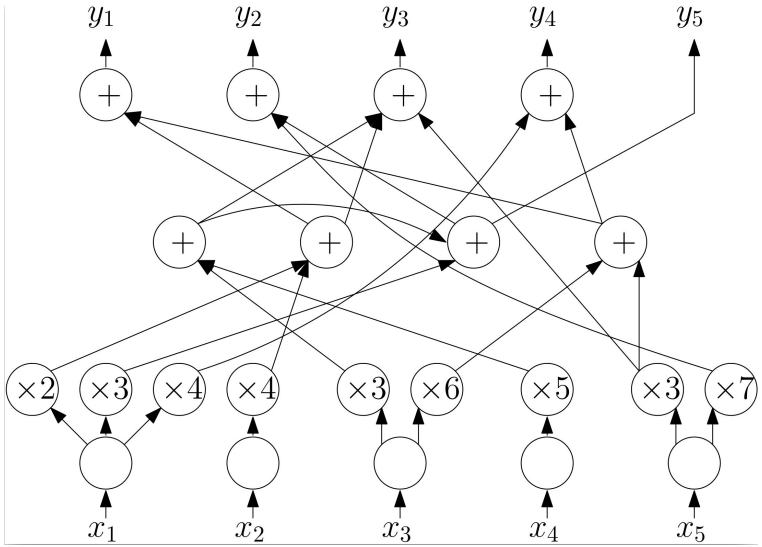
$$\begin{bmatrix} 1.2 & 3.4 & 5.6 & 0 & 2.3 \\ 2.3 & 0 & 2.3 & 4.5 & 1.7 \\ 1.2 & 3.4 & 2.3 & 4.5 & 0 \\ 3.4 & 0 & 5.6 & 0 & 2.3 \\ 2.3 & 0 & 2.3 & 4.5 & 0 \\ 1.2 & 3.4 & 2.3 & 4.5 & 3.4 \end{bmatrix}$$

$$V = [1.2 \quad 1.7 \quad 2.3 \quad 3.4 \quad 4.5 \quad 5.6]$$

$$\begin{aligned} S = {} & A_{1,1}\ A_{4,2}\ A_{6,3}\ A_{3,5}\ Z\ A_{3,1}\ A_{3,3}\ A_{5,4}\ A_{2,5}\ Z \\ & A_{1,1}\ A_{4,2}\ A_{3,3}\ A_{5,4}\ Z\ A_{4,1}\ A_{6,3}\ A_{3,5}\ Z \\ & A_{3,1}\ A_{3,3}\ A_{5,4}\ Z\ A_{1,1}\ A_{4,2}\ A_{3,3}\ A_{5,4}\ A_{4,5}\ Z \end{aligned}$$

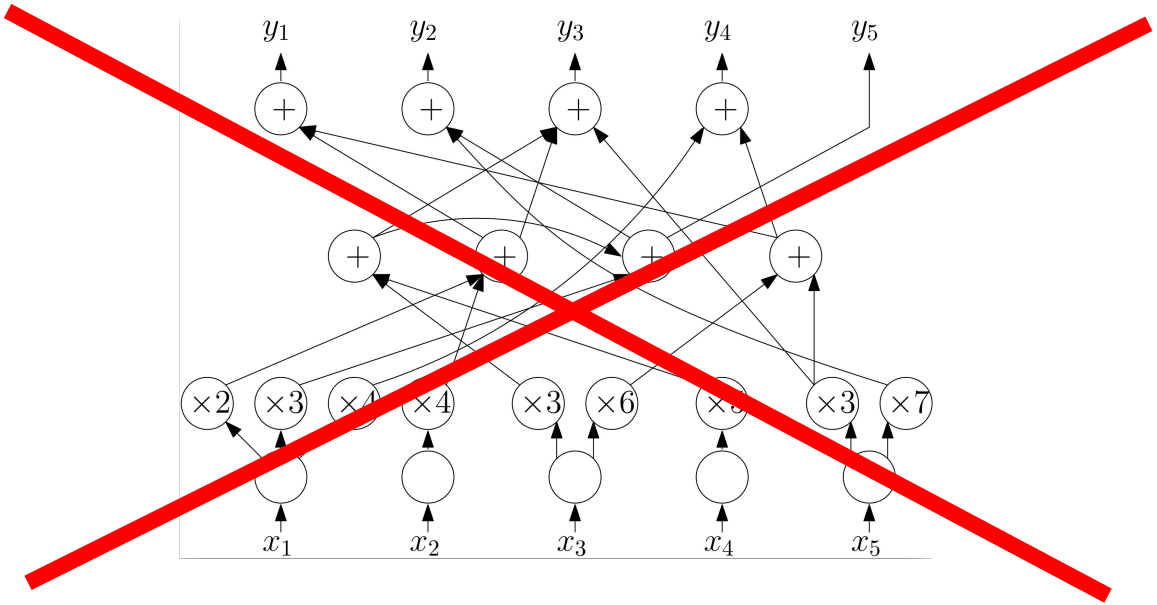
Figure 1: A matrix and its value/column representation. In the string S the symbol $A_{3,1}$ stands for an occurrence of the value $V[3] = 2.3$ in column 1. Note that the same value in column 3, is represent instead by $A_{3,3}$.

Basic idea:



Use a grammar to compress the matrix and transform the grammar into an arithmetic circuit to compute the matrix vector multiplication

Some progress:



We simplified the algorithm getting rid of the arithmetic circuit

Our competitor: CLA (VLDBJ, CACM)

- Does not output the compressed representation (recomputed at every execution)
- With a 30GB heap fails to process a matrix with 1.5GB nonzeros
- The measured compression ratios differ from the ones reported in the paper
- As the computation goes on, the memory usage grows and the program becomes very slow…

We have contacted the authors trying to address the above issues and ensure a fair comparison

Compression results (preliminary)

Dataset	rows	cols	% nonzero	RePair	gzip	CLA
Higgs	11,000,000	28	0.92	1,020	1,039	2,362
Census	2,458,285	68	0.43	27	45	54
Covtype	581,012	54	0.22	11	14	13
Mnist2m	2,000,000	784	0.25	732	1,036	699

Similar matrix multiplication times (secs for large matrices, millisecs for small ones)

Next steps

- Improve compression & running time (with unipi)
- Experiments with NN matrices (with unimi)

Questions?